



Stacy Stolen Code

Forensic Analysis

Prepared By

Facundo Lerena

Contents of this document

Contents of this document	2
Context of this Document	3
Timeline (non-exhaustive)	4
Evidence	5
I. First	5
II. Second	5
III. Third	5
IV. Fourth	6
V. Fifth	7
VI. Sixth	7
VII. Seventh	7

Context of this Document

Last week, we presented our tool [Stacy](#) on X, and we received a reply from a community member asking about the high-level differences between our tool and the one presented by the hackathon winners, GeckoSec, also known as Jeevan Jutla. This prompted us to investigate further and find out about the misappropriation of the entirety of our code, without adding any new features nor respecting our MIT license.

Timeline (non-exhaustive)

(Timestamps are UTC - 0)

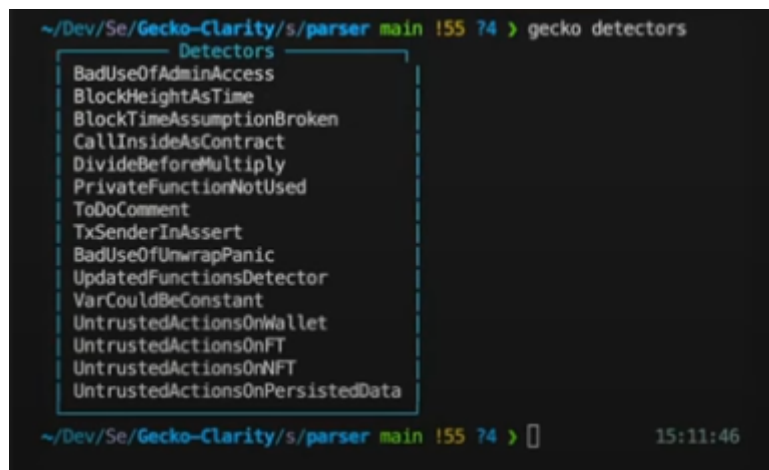
- **15:30, May 16, 2024** - CoinFabrik <> Stacks meeting to talk about fuzzing and static analysis.
- **15:10, May 23, 2024** - Stacy code submitted in a public repository from [Facundo Lerena](#) developer @ CoinFabrik to test GitHub actions: [\[link at commit\]](#)
- **May 29 - May 31, 2024** - [EasyA Consensus hackathon](#) - Gecko wins first place. This, as proved below, makes Gecko violate the [Terms from the hackathon](#), specifically Rule 8.
- **May 30, 2024** - [Commit from @jjjutla](#) removing links to [CoinFabrik's website](#).
- **16:03, May 31, 2024**: [@gecko_sec post about their analyzer](#)
 - a. The repository posted there, at 19:24 Jul 9, 2024 throws 404: [\[link\]](#). We strongly believe the repo had a name change in between that date and today, and this is the repository we are talking about: [\[link\]](#)
- **14:10:42, Jun 3, 2024**: [Facundo Lerena](#) commits on [CoinFabrik/stacy](#) the code from his repository: [\[link\]](#)
- **15:30, Jun 12, 2024**: Meeting CoinFabrik <> Stacks to talk about Stacy.
- **15:04 PM, Jun 19, 2024**: Gecko [gets a client](#) using the copied code.
- **Jul 2, 2024**: Eric Choy from Hiro [hosts an episode of Web3 On Bitcoin Explained with @jjjutla](#). They show their static analyzer, with a bug present in Facundo Lerena's repo @ 8:25. This is further explained in the [Evidence section](#).
- **18:19, Jul 8, 2024**: [@herogamer21btc asks in X](#) "What's the high level differences to @gecko_sec" in our [post](#). This was the post that made us start this investigation.
- **18:56, Jul 6, 2024**: Jeevan Jutla opens an issue in [hirosystems/explorer](#) claiming that they developed the static analyzer. At 13:36, Jul 8, 2024 [@andresgalante responded to the issue](#).

Evidence

I. First

As an error, we copied some templating code for publishing in pipy, and in that we left a `Number` class that is not used, and it's not even necessary in Stacy. Gecko [even copied](#) that class as a `struct` in Rust. We removed that a [few days ago](#). Also, [here it checks](#) for a file called `__init__.rs`, a file that does not exist in Rust and it's a copy of our [check to ignore that file](#).

II. Second



```
~/Dev/Se/Gecko-Clarity/s/parser main !55 74 > gecko detectors
Detectors
BadUseOfAdminAccess
BlockHeightAsTime
BlockTimeAssumptionBroken
CallInsideAsContract
DivideBeforeMultiply
PrivateFunctionNotUsed
ToDoComment
TxSenderInAssert
BadUseOfUnwrapPanic
UpdatedFunctionsDetector
VarCouldBeConstant
UntrustedActionsOnWallet
UntrustedActionsOnNFT
UntrustedActionsOnNFT
UntrustedActionsOnPersistedData
~/Dev/Se/Gecko-Clarity/s/parser main !55 74 > 15:11:46
```

The upper right corner has a bug in the number of line characters printed, and the detector names (except a few) are exactly the same. This bug is replicable in [Facundo Lerena](#)'s repository.

III. Third

The (MIT) LICENSE file was pushed to [faculerena/stacy](#) with

```
`Copyright (c) 2024 Coinfabrik` in the LICENSE file.
```

```
commit cffab32acd1d62dea75e2b22b2d2d1e87a097a83
Author: Camila Gallo <camilagallo1700@gmail.com>
Date: Thu May 30 16:22:16 2024
```


[Gecko-Security/gecko-stx](#) does not have a LICENSE file, and in this commit they started to remove references to CoinFabrik.

```
commit 4e39f4c2a5155b6bc9f756aa83a143e3e43b6f26
Author: jjjutla <jeevan.jutla@gmail.com>
Date: Thu May 30 18:45:17 2024
```

Update README.md

The following commands were ran to get the timestamps in UTC-0, the first one in [faculereana/stacy](#), and the second one in [Gecko-Security/gecko-stx](#)

```
TZ=UTC0 git log --date=local | grep
"cffab32acd1d62dea75e2b22b2d2d1e87a097a83" -C 5
```

```
TZ=UTC0 git log --date=local | grep
"4e39f4c2a5155b6bc9f756aa83a143e3e43b6f26" -C 5
```

IV. Fourth

How Gecko's Static Analysis Works

1. We parse the Clarity code into a structure that Gecko can understand, this is called an Abstract Syntax Tree (AST). It represents the hierarchical structure of the code. We use the [Clarity Contract Analysis Crate](#), which converts Clarity code into an AST and other metadata. This is the main entrypoint for Gecko.
2. We then define a struct called [Gecko](#), which implements the `ast_visitor` crate used to traverse each node and understand the behaviour of the code.
3. Taint analysis is used to track the flow of potentially unsafe data through the program and locate bugs and vulnearbilities. This involves defining the vulnearbility detectors as invariants and tracking the data to ensure it is properly checked or sanitized.
4. As Gecko traverses the tree it propagates this taint to other nodes that depend on these.
5. Once the traversal is complete messages are displayed about issues found including the location of the bug in source.

[jeevan jutla](#) claims in the [readme](#) that Gecko implements ``ast_visitor``. This is not how the code works. The ``ast_visitor`` link goes to the rust's `ast`Visitor`` trait, which cannot be implemented over Clarity's AST, as it's targeted for Rust code. We used that name as we previously worked on [Scout](#), and we took inspiration in the architecture of Clippy (from Rust) to create Stacy.

V. Fifth

The [six detectors](#) that Gecko-stx currently implements effectively have the same names as the first iteration in [faculerena/Stacy at the moment of the hackathon](#).

VI. Sixth

The only tree-sitter grammar written for clarity was created by [Franco Bregante](#), auditor at CoinFabrik, a few years ago. The generated files that are exposed in the [parser folder](#) seem to have small differences, but are based on the [xlittlerag/tree-sitter-clarity](#) grammar for Clarity.

VII. Seventh

At 18:11:36, Jul 2, 2024, @jjjutla commits to [Gecko-Security/gecko-stx](#) a [copy](#) of [this file](#), an [ltyFuzz](#) core file, with some small surface level changes (renaming and comment deletion).